

作業の効率化でがんばってます

JANOG 45 Lightning Talk (初心者枠 1位)

株式会社Ping-t
谷岡 英治(CCIE #52766)

自己紹介

- ・ 谷岡 英治 (CCIE #52766)
- ・ 株式会社Ping-t (<https://ping-t.com/>)
- ・ JANOG登壇経験 : 3回め
 - ・ JANOG42 (津) LT...「IT系資格の取得について考えてみた」
 - ・ JANOG43 (甲府) 初心者LT...「JANOGスタッフやってみて」
 - ・ **【今回】**JANOG45 LT[初心者枠]...? 🤔
- ・ 今回はIgnite Talks形式にチャレンジします！

Ignite Talksってなに？



IGNITE IS A SERIES OF SPEEDY PRESENTATIONS

Presenters get 20 slides, which automatically advance every 15 seconds. The result is a fast and fun presentation which lasts just 5 minutes.

Ignite events are held in cities around the world.

- <https://www.ignitetalks.io/>
 - 20スライド固定
 - 各スライド15秒で強制的に切り替え

なんでIgnite Talks ?

- JANOG44の当日LTでぺちゃくちゃ（20スライド固定、各スライド20秒）形式をやっていた。面白そう！
 - 20秒もしゃべれない…
 - LTだから5分に収めたい…
- Ignite Talksなら行けるかも？
 - $20 \times 15 \text{秒} = 300 \text{秒} = 5 \text{分}$ ！
 - 暖かく見守ってください…

普段の業務

- ・ 資格試験学習用の問題、解説の作成、レビュー
- ・ **頻繁な実機検証**
 - ・ 作っては壊し、作っては壊し
 - ・ 似たような環境を何度も作ったり
 - ・ でも、保存したとしてもそんなに再利用しない

毎回さくさくやれるようにしよう！

作業環境の前提

- Cisco試験、LPIC/LinuC試験がメイン
 - CLI操作が主体
- 作業環境はmacOS
 - Windowsはいいシェルがなくて困る（愚痴）
 - いまだとNYAGOSかな
<https://github.com/zetamatta/nyagos>

作業の効率化に必要なのは？

- ・ 時間短縮
- ・ 並列実行
- ・ ミス削減

時間短縮の例

- ・ 機材準備不要
- ・ 機械的（自動化）

時間短縮に使っているもの（機材準備不要）

- GNS3（シミュレータ）
 - 機器の準備、物理結線が不要
 - 事前に構成を作っておくのも可能
 - ネットワーク機器（IOSなど）だけではなく、DockerイメージやVMも扱える
 - 物理ネットワークとも接続可能
- EVE-NG（シミュレータ）
 - GNS3と同様
 - Web-UIなので、ローカルへのインストールが不要

時間短縮に使っているもの（機械的）

- 期待する初期状態にすぐに持っていく

- IOS

#config replace flash:default.cfg force

※事前に「#copy run flash:default.cfg」しておく

- IOS-XR

#configure

(config)#load disk0:default.cfg

(config)#commit replace

※事前に「(config)#save configuration disk0:default.cfg」しておく

並列実行の例

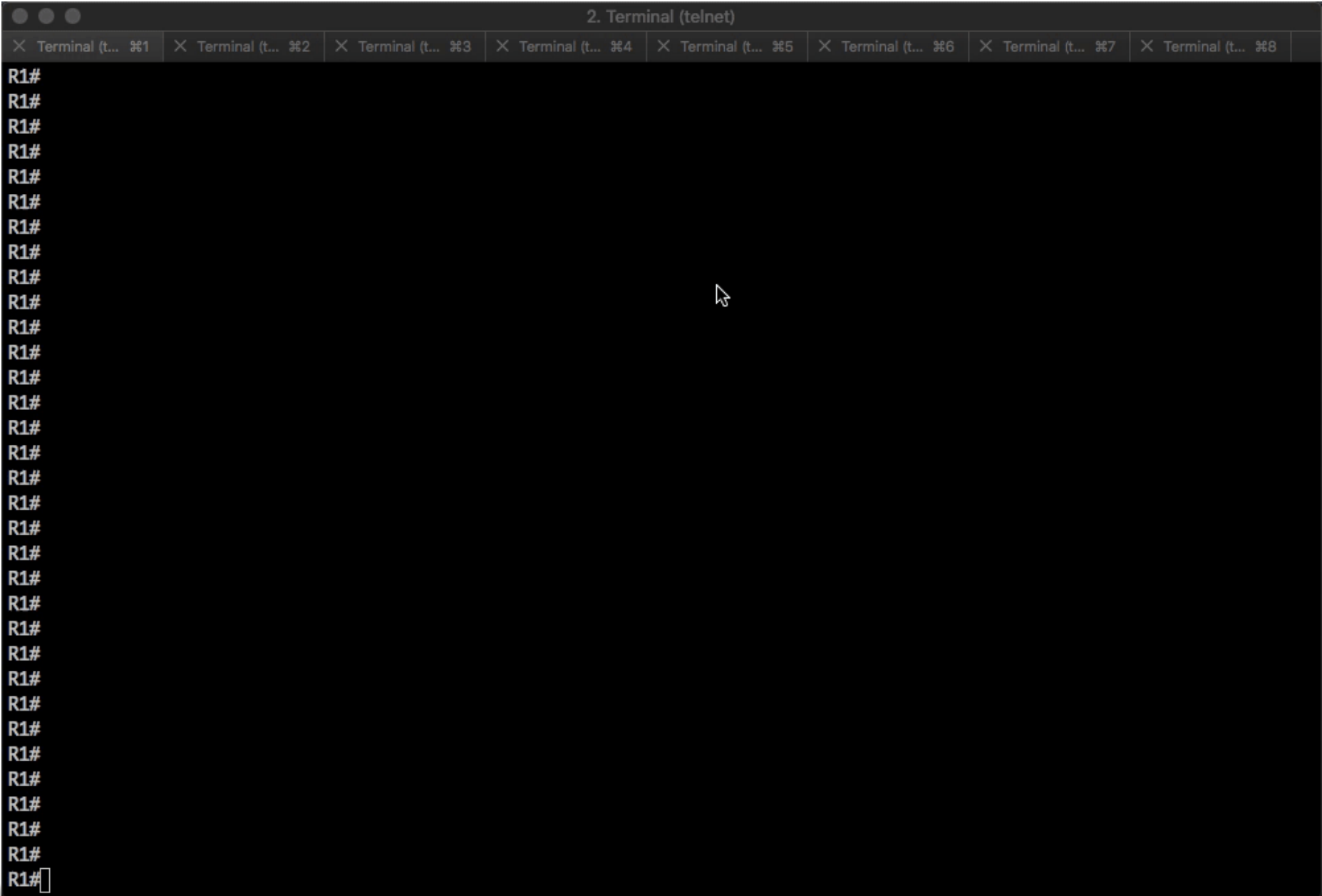
- ・ 同時コマンド投入
- ・ 同時編集

並列実行（同時コマンド投入）に使っているもの

- iTerm2(ターミナル)
 - 1ウィンドウに複数のタブ＝同時にコンソール接続できる
 - 複数のタブに同時に同じ入力
 - Broadcast Input（⌘I）
 - 全ルータで同じ設定入れるときとか

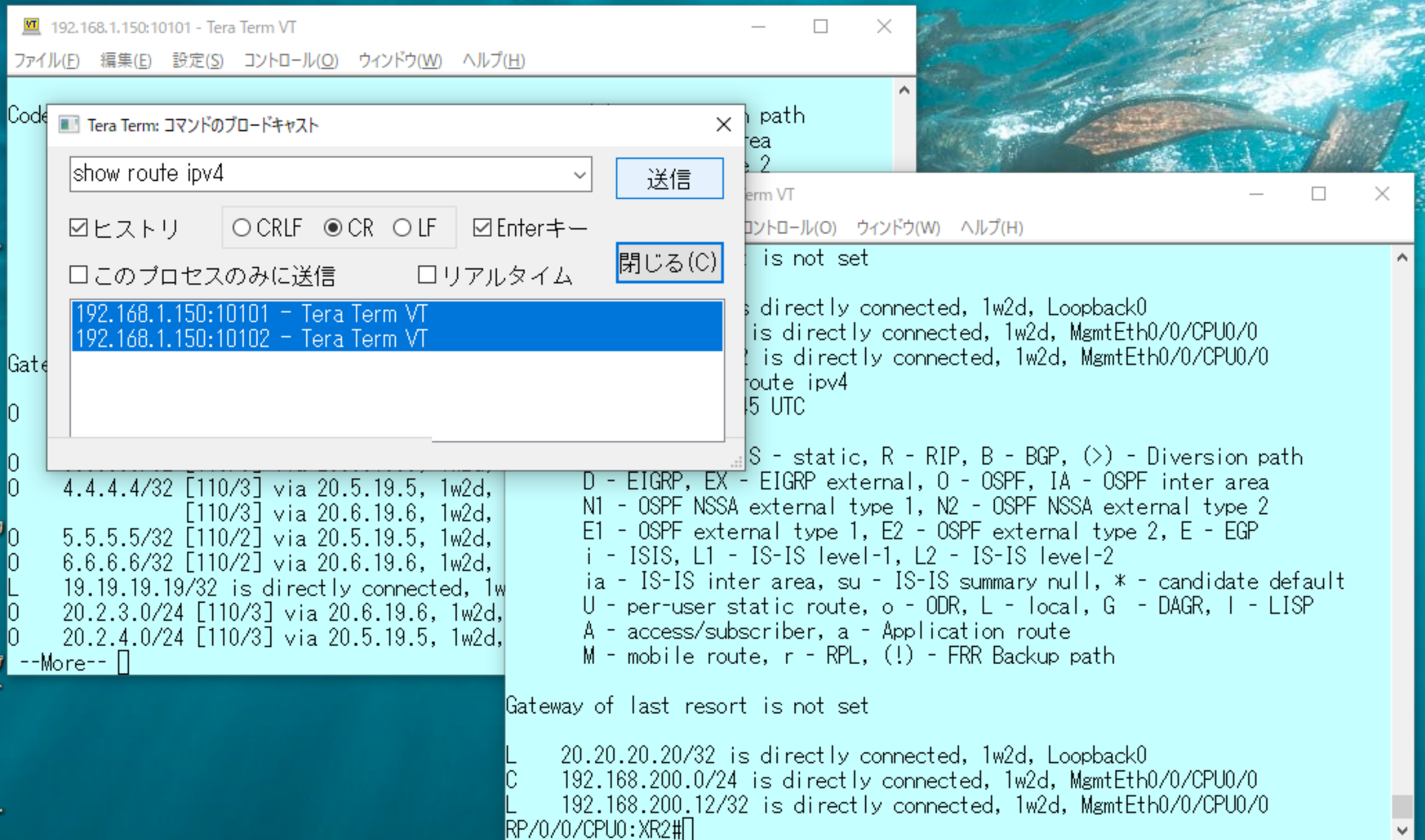
```
mpls ldp router-id Loopback0
router ospf 1
mpls ldp autoconfig
```

実行例



自分は普段は使っていないけど

- ・ 皆様ご存知TeraTermにも同じ機能が。
 - ・ ブロードキャストコマンド！



Tera Term: コマンドのブロードキャスト

show route ipv4

送信

☒ ヒストリ ☐ CRLF ☒ CR ☐ LF ☒ Enterキー

☐ このプロセスのみに送信 ☐ リアルタイム

閉じる(C)

- 192.168.1.150:10101 - Tera Term VT
- 192.168.1.150:10102 - Tera Term VT

Gate

0

0

0 4.4.4.4/32 [110/3] via 20.5.19.5, 1w2d,
[110/3] via 20.6.19.6, 1w2d,
0 5.5.5.5/32 [110/2] via 20.5.19.5, 1w2d,
0 6.6.6.6/32 [110/2] via 20.6.19.6, 1w2d,
L 19.19.19.19/32 is directly connected, 1w2d,
0 20.2.3.0/24 [110/3] via 20.6.19.6, 1w2d,
0 20.2.4.0/24 [110/3] via 20.5.19.5, 1w2d,
--More--

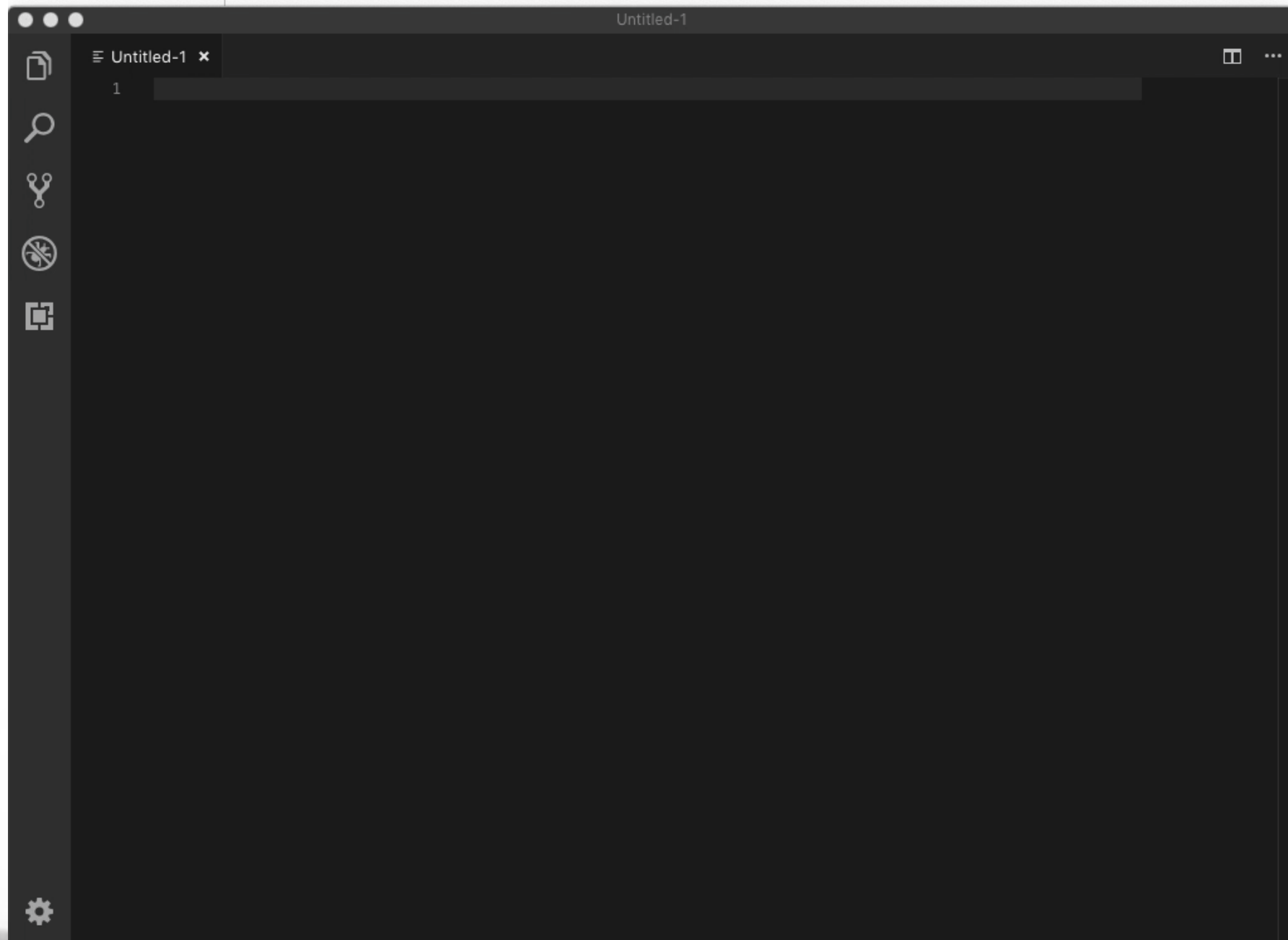
Gateway of last resort is not set

L 20.20.20.20/32 is directly connected, 1w2d, Loopback0
C 192.168.200.0/24 is directly connected, 1w2d, MgmtEth0/0/CPU0/0
L 192.168.200.12/32 is directly connected, 1w2d, MgmtEth0/0/CPU0/0
RP/0/0/CPU0:XR2#

並列実行（同時編集）に使っているもの

- Visual Studio Code
 - エディタ
 - 同じ文字列を選択：⌘+D、検索後⌘+クリック
 - ⌘ + Option + ↑ ↓ （カーソルキー）
 - 選択した範囲を同時に編集

実行例



ミス削減の例

- プログラム化 = 自動化
 - 結果のチェックを目grepしない
 - 出力を必要なものだけに絞る

ミス削減に使っているもの

- 環境構築シェルスクリプト
 - 例) GNS3上のDockerインスタンスの環境を整える
 - 入れたいパッケージ
 - サービス起動

効率化できてる？

- できてる！
- さらに
 - GNS3→ローカルにインストールしとけば、どこでも作業できる
 - iTerm2→多機能なのでBroadcast Input以外にも使える
 - VScode→プログラム書くときも役に立つ

まとめ

使えれば、何でも使って、効率化！

→普段使っていない機能、探すと役に立つものがあるかも？